WHAT IS CLAIMED IS:

Sub B1

1. A method for the computer animation of two colliding bodies by the iterative calculation of body positions from previous positions, said method comprising:

receiving data defining physical properties of said bodies, said physical properties including positions and shapes of said bodies;

determining a collision between said bodies;

calculating a closest-points vector between said two bodies from said physical properties of said bodies;

calculating a contact force on at least one of said bodies, said contact force along said closest-points vector and having a nonlinear relationship with respect to said closest-points vector such that said force increases sufficiently rapidly as said closest-points vector goes to zero to overcome motions causing said collision between said bodies;

calculating a position of said at least one of said bodies at a subsequent time interval in response to said contact force upon said body;

iteratively repeating said collision determining, closest-points vector calculating, contact force calculating and position calculating steps for subsequent time intervals; and

displaying calculated positions of said bodies at selected time intervals for a realistic animation of said colliding bodies.

Sub C2

2. The method of claim 1 wherein said contact force F has the relationship:

$$F = ae^{b(c-x)}-d,$$

where a, b, c, and d are constants, x is the length of the closest-points vector.

3. The method of claim 1 wherein said contact force F has the relationship:

$$F = a/(b-x)$$

where a, and b are constants, and x is the length of the closest-points vector.

4. The method of claim 1 wherein said bodies are defined with incompressible portions, said closest-points vector calculated between said incompressible portions of said two bodies.

5. The method of claim 4 wherein said bodies are defined with compressible portions covering said incompressible portions, said collision determining step comprising determining whether said compressible portions of said bodies are in contact.

Sub B^2

6. The method of claim 1 wherein at least one of said colliding bodies comprises a first body part connected to a second body part by a joint, said joint having at least one rotational degree of freedom, a joint limit stop and a maximum allowed rotation limit; and comprising:

determining a rotation of said first body part about said joint to reach said joint limit stop;

calculating a torque upon said first body part along said one rotational degree of freedom, said torque having a nonlinear relationship with respect to an angle of rotation from said maximum allowed rotation limit such that said torque increases sufficiently rapidly as said angle goes to zero to overcome motions causing said rotation of said first body part about said joint;

calculating a rotational position of said body part at a subsequent time interval in response to said torque upon said body part; and

iteratively repeating said rotation determining, said torque calculating, and said rotational position calculating steps for subsequent time intervals.

7. The method of claim 6 wherein said torque T has the relationship:

$$T = ae^{b(c-\theta)}-d,$$

where a, b, c, and d are constants, and θ is an angle of said body part from said maximum allowed rotation limit.

8. The method of claim 6 wherein said torque T has the relationship:

$$T = a/(b-\theta)$$

where a and b are constants, θ is an angle of said body part from said maximum allowed rotation limit.

9. The method of claim 1 further comprising:

determining a fault in any one of said closest-points vector, contact force, and position calculating steps;

reinitializing said faulty calculating step; and

performing said reinitialized calculating step with a smaller time interval.

10. The method of claim 9 further comprising:

repeating said fault determining, reinitializing and reinitialized calculating step performing steps at smaller and smaller time intervals until said fault is eliminated.

11. The method of claim 9 wherein said smaller time interval is one-half of a previous time interval.

12. The method of claim 9 wherein said fault comprises floating-point numeric overflow.

13. The method of claim 9 wherein said fault comprises said closest-points vector being less than zero.

14. The method of claim 9 wherein said fault comprises physically incorrect collision situations involving excessive interpenetration of bodies.

15. The method of claim 6 further comprising:

determining a fault in any one of said closest-points vector, contact force, position, torque and rotational position calculating steps;

reinitializing said faulty calculating step; and

performing said reinitialized calculating step with a smaller time interval.

16. The method of claim 15 wherein said joint comprises a gimbal joint and said fault comprises said gimbal joint reaching a gimbal lock singularity.

17. The method of claim 15 wherein said rotational position calculating step comprises calculating Euler parameters and said fault comprises non-normalized Euler parameters.

18 The method of claim 15 wherein said fault comprises first body part rotates beyond said maximum allowed rotation limit.

Sub B^3

19. A computer program for the realistic computer animation of two colliding bodies by the iterative calculation of body positions from previous positions, comprising:

code that receives data defining physical properties of said bodies, said physical properties including positions and shapes of said bodies;

code that determines a collision between said bodies;

code that calculates a closest-points vector between said two bodies from said physical properties of said bodies;

code that calculates a contact force on at least one of said bodies, said contact force along said closest-points vector and having a nonlinear relationship with respect to said closest-points vector such that said force increases sufficiently rapidly as said closest-points vector goes to zero to overcome motions causing said collision between said bodies;

code that calculates a position of said at least one of said bodies at a subsequent time interval in response to said contact force upon said body;

code that iteratively repeats said collision determining, closest-points vector calculating, contact force calculating and position calculating steps for subsequent time intervals;

code that allows a computer display to display calculated positions of said bodies at selected time intervals for a realistic animation of said colliding bodies; and

a computer readable medium that stores said codes.

Sub C4

20. The computer program of claim 19 wherein said contact force ~~F has~~ the relationship:

$$F = ae^{b(c-x)} - d,$$

where a, b, c, ~~and~~ d are constants, x is the length of the closest-points vector.

21. The computer program of claim 19 wherein said contact force F has the relationship:

$$F = a/(b-x)$$

where a, and b are constants, and x is the length of the closest-points vector.

22. The computer program of claim 19 wherein said bodies are defined with incompressible portions, said closest-points vector calculated between said incompressible portions of said two bodies.

23. The computer program of claim 22 wherein said bodies are defined with compressible portions covering said incompressible portions, said collision determining step comprises code that determines whether said compressible portions of said bodies are in contact.

Sub B4

24. The computer program of claim 19 wherein at least one of said colliding bodies is defined as a first body part connected to a second body part by a joint, said joint having at least one rotational degree of freedom, a joint limit stop and a maximum allowed rotation limit; and said computer program comprising:

code that determines a rotation of said first body part about said joint to reach said joint limit stop;

code that calculates a torque upon said first body part along said one rotational degree of freedom, said torque having a nonlinear relationship with respect to an angle of rotation from said maximum allowed rotation limit such that said torque increases sufficiently rapidly as said angle goes to zero to overcome motions causing said rotation of said first body part about said joint;

code that calculates a rotational position of said body part at a subsequent time interval in response to said torque upon said body part; and

code that iteratively repeats said rotation determining, said torque calculating, and said rotational position calculating steps for subsequent time intervals.

25. The computer program of claim 24 wherein said torque T has the relationship:

$$T = ae^{b(c-\theta)}-d,$$

where a, b, c, and d are constants, and θ is an angle of said body part from said maximum allowed rotation limit.

26. The method of claim 24 wherein said torque T has the relationship:

$$T = a/(b-\theta)$$

where a and b are constants, θ is an angle of said body part from said maximum allowed rotation limit.

27. The computer program of claim 19 further comprising:

code that determines a fault in an operation of any one of said closest-points vector, contact force, and position calculating codes;

code that reinitializes said faulty operation; and

code that performs said reinitialized operation with a smaller time interval.

28. The computer program of claim 27 further comprising:

code that repeats operations of said fault determining, reinitializing and reinitialized operation performing codes at smaller and smaller time intervals until said fault is eliminated.

29. The computer program of claim 27 wherein said smaller time interval is one-half of a previous time interval.

30. The computer program of claim 27 wherein said fault comprises floating-point numeric overflow.

31. The computer program of claim 27 wherein said fault comprises said closest-points vector being less than zero.

32. The computer program of claim 27 wherein said fault comprises physically incorrect collision situations involving excessive interpenetration of bodies.

33. The computer program of claim 24 further comprising:

code that determines a fault in an operation of any one of said closest-points vector, contact force, position, torque and rotational position calculating codes;

code that reinitializes said faulty operation; and

code that performs said reinitialized operation with a smaller time interval.

34. The computer program of claim 33 wherein said joint comprises a gimbal joint and said fault comprises said gimbal joint reaching a gimbal lock singularity.

35. The computer program of claim 33 wherein operation of said rotational position calculating code comprises calculating Euler parameters and said fault comprises non-normalized Euler parameters.

36. The computer program of claim ~~15~~ wherein said fault comprises first body part rotating beyond said ~~maximum~~ allowed rotation limit.